SOMNIUM VR1

Frequently Asked Questions

Somnium VR1 - Frequently Asked Question

- Should I use SteamVR or OpenXR?

This will depend on how the end application (game, flight/racing sim) connects to the headset. Mainly there are two options:

1) Application which uses OpenVR/SteamVR API:

- This means that the application will always connect to SteamVR, and it's up to SteamVR to provide connection to the headset.
- SteamVR then uses custom drivers specific for each headset.
- For Somnium VR1 we provide our "vrg" SteamVR driver (or "somnium" for the new beta).
- This is installed directly to SteamVR/drivers/ folder so SteamVR automatically recognize it, and so we don't edit the openvrpaths.vrpath file.
- You can freely choose which drivers will be enabled or disabled by accessing the main SteamVR menu - Startup/Shutdown - Manage Add-ons.
- You can also disable/enable the specific SteamVR drivers from their own service apps (so Somnium VR1 Tool for Somnium headset), but this behaves the same as disabling/enabling them in the SteamVR menu.

2) application which uses OpenXR API:

2a) we recommend using our custom OpenXR runtime for that

- This means that the application connects directly to specific OpenXR runtime, so it doesn't need SteamVR (except for access to tracking data mentioned below).
- Only one OpenXR runtime can be active on the PC at one time, and this is controlled by changing the registry key "HKEY_LOCAL_MACHINE\SOFTWARE\Khronos\OpenXR\1\ActiveRuntime".
- For Somnium we provide our "vrg_openxr" runtime (or "somnium_openxr" for the new beta).
- To enable the runtime (i.e., set it as currently active OpenXR runtime on the PC) you can use the OpenXR - Enable switch in our Somnium VR1 Tool service application.
- When running an OpenXR application with our custom OpenXR runtime, SteamVR will be accessed in the background to get tracking data from lighthouse base stations.
- If you also have vrg/somnium SteamVR driver enabled, you should make sure that SteamVR is turned off first before starting the end application (game, flight/racing sim).

2b) as an alternative you can utilize the SteamVR OpenXR runtime

- This OpenXR runtime is provided as a part of standard SteamVR installation.
- It creates a bridge between SteamVR software and OpenXR compatible applications, so you can use this runtime along with the SteamVR driver for specific headset mentioned in 1) even if the end application doesn't support OpenVR/SteamVR API directly.
- So, for this approach to work with Somnium VR1 you'll also need to enable the vrg/somnium SteamVR driver.
- You can enable the SteamVR OpenXR runtime in the main SteamVR menu - OpenXR - "Set SteamVR as OpenXR runtime".
- Enabling this OpenXR runtime will disable our custom vrg/somnium OpenXR runtime, and vice versa, as only one can be active at any time.

- How can I enable foveated rendering?

Somnium VR1 runtime supports foveated rendering based on separate textures/views for background and foveated area.

This technique is also called quad views, because in total the application renders 4 different views:
- Background view for the left eye.
- Background view for the right eye.
- Foveated area for the left eye.
- Foveated area for the right eye.

You can customize the resolution of the background, resolution of the foveated area, and size of the foveated area, in Somnium VR1 Tool (Settings - Headset - Rendering page).

To be able to utilize this, there are two main requirements:
- You need to use our custom vrg/somnium OpenXR runtime (as described in the point 2a in "Should I use SteamVR or OpenXR").
- The end application (game, flight/racing sim) needs to support OpenXR implementation for foveated views (XR_VIEW_CONFIGURATION_TYPE_PRIMARY_STEREO_WITH_FOVEATED_INSET or previously XR_VIEW_CONFIGURATION_TYPE_PRIMARY_QUAD_VARJO).
- See OpenXR specification for more details (https://registry.khronos.org/OpenXR/specs/1.1/html/xrspec.html#XrViewConfigurationType).
- If the end application uses older OpenXR API (1.0) it also needs to support the "XR_VARJO_quad_views" extension.
- Currently know applications which support this: DCS World, Unreal Engine 5, Pavlov VR, Unity (requires additional custom script).

Then depending on the state/availability of eye tracking we differentiate:

1) Fixed/static foveated rendering
- It's available even if eye tracking is not calibrated or not supported by the headset
- The foveated area will be placed in approximately middle of the view (depending on the FOV and resolution values).

2) Dynamic foveated rendering
- Requires headset with eye tracking support.
- Eye tracking has to be calibrated first using the Eye Tracking Calibration button in Somnium VR1 Tool (if eye tracking is not yet calibrated fixed/static foveated rendering will be used).
- We recommend using the "Advanced" eye tracking calibration for more precise eye tracking results
- If the end application uses older OpenXR API (1.0) it also needs to support the "XR_VARJO_foveated_rendering" extension.

- Is mbuchia's Quad-Views-Foveated extension compatible with Somnium VR1?

What the Quad-Views-Foveated extension does is that it converts the quad views data from the end application to standard stereo rendering views for the headset runtime. This allows runtimes without quad views support to utilize this functionality.

So in principle yes this should be compatible with Somnium VR1, but we don't recommend using it as our vrg/somnium OpenXR runtime already has the quad views functionality integrated.
Using this extension will still only allow you to use foveated rendering in end applications which support the OpenXR quad views implementation (see the list of application in "How can I enable foveated rendering?")

- How can I use hand tracking?

Hand tracking variants of Somnium VR1 headsets are equipped with embedded version of Ultraleap hand tracking sensor, so any application which supports Ultraleap hand tracking will also be directly compatible with Somnium VR1.

There are several options on how to utilize the Ultraleap hand tracking in a VR application:

1) Direct support for Ultraleap API

- If the end application integrates Ultraleap API it can connect directly to Ultraleap runtime (installed from Somnium VR1 Tool).
- This approach is independent on the used headset or VR API (SteamVR, OpenXR).
- This is supported for example in Ultraleap demos from their website (Blocks, Cat Explorer, etc.) or in most game engines (Unreal Engine, Unity).

2) Support for OpenXR hand tracking functionality

- If the end application supports OpenXR hand tracking extension (XR_EXT_hand_tracking) it can utilize any hand tracking data provided by OpenXR runtime.
- If you are using our vrg/somnium OpenXR runtime, it provides the hand tracking data directly.
- If you are using SteamVR OpenXR runtime you can utilize the Ultraleap OpenXR API layer, which is provided as part of the Ultraleap installation.

3) SteamVR add-ons

- Even if the application doesn't support hand tracking directly, it can still be utilized as a standard VR controller.
- You will need a third-party add-on for SteamVR which masks Ultraleap hand tracking data as VR controller (for example: https://github.com/SDraw/driver_leap).

4) Somnium hand tracking overlay visualization

- Somnium VR1 Tool also provides a simple way to visualize tracked hands in any applications.
- This can be enabled in Settings - Headset - Hands tab with the "Show Hands" switch.
- A simplified 3D model of the virtual hands will be shown as overlay inside the headset.
- To also show this overlay on the desktop you'll need to enable the VR View window.
- This approach doesn't allow any hand tracking interactions with the end applications, and it's only for visualization.

- Is OpenXR Toolkit compatible with Somnium VR1?

No, we don't officially support OpenXR Toolkit, as it was recently discontinued and it's no longer supported by its' developers. It can still be beneficial for some end applications, as it supports some functionality which is not originally available there, but we don't include it in our compatibility testing.

- Which reprojection options are supported by Somnium VR1?

Asynchronous reprojection is a technique aimed to make the VR image more responsive, even if FPS in the end application can't match the frequency of the headset displays. There are several different types and implementations of asynchronous reprojection and what is available will depend on the used end application and VR API.

1) SteamVR

- If you are using Somnium VR1 through our SteamVR driver, you can utilize all standard functionality of SteamVR.
- The main algorithm for reprojection in SteamVR pipeline is called motion smoothing, and it will automatically limit FPS to half the refresh rate if it can't match the full refresh rate of the displays.
- This is supported in all SteamVR compatible applications by default, as it doesn't require any additional implementation on the application side.
- These options can be disabled in SteamVR settings.

2) OpenXR runtime

- If you are using our custom vrg/somnium OpenXR runtime, you can utilize two different algorithms for reprojection called "timewarp" and "spacewarp".

Timewarp

- It uses rotational data (3DoF) to predict changes in rotation between the time when the image is originally rendered and when it is displayed inside the headset.
- It is supported by al OpenXR applications by default, as it doesn't require any additional implementation.
- It can't compensate for translation changes (6DoF) so for example if you move forward/backward you will still be able to notice the lower framerate.

Spacewarp

- It uses full 3D depth map of the virtual scene to also compensate for translation changes (6DoF)
- It requires the end application to submit depth map data with each frame (using XR_KHR_composition_layer_depth extension).
- Currently know applications which support this: MSFS, Unreal Engine 5, Unity.

- If I encounter any issues with the Somnium VR1 Tool, software, or headset itself, which files should I sent to Somnium support?

You should always send log files from "C:/ProgramData/vrg/" folder, specifically:

- vrgeyes.log
- vrgloader.log
- vrgsdk.log
- old_logs folder
- tool_logs folder

If this issue was encountered while using the SteamVR driver, you should also send us System Report file from SteamVR:

- open SteamVR.
- select "Create System Report" from the main SteamVR menu.
- in the new window select "Save to File".

If the end application also has its' own logs, you should send those as well.

- for example, for DCS you can find the dcs.log file in "C:\Users\username\Saved Games\DCS\Logs"

- How can I fix the NVAPI_DISPLAY_IN_USE error while trying to run an OpenXR application?

If you see the "NVAPI_DISPLAY_IN_USE" error, it means that two different applications or VR APIs are trying to initialize the headset at the same time.

Most often this happens if SteamVR is already rendering image into Somnium headset, and you try to start another application through our OpenXR runtime.

To fix this issue:

- Make sure that SteamVR is not running before you start the OpenXR application.
- If the issue persists, try disabling the SteamVR driver completely (either through Somnium VR Tool in Settings - Software - Drivers page, or through SteamVR in Startup/Shutdown - Manage Add-ons).
- If you want to run the application through SteamVR instead, either make sure that the application supports OpenVR API, or enable the OpenXR runtime included in SteamVR (in main SteamVR menu - OpenXR - "Set SteamVR as OpenXR runtime").
- See details in "Should I use SteamVR or OpenXR?".

- Why can't I enable or calibrate eye tracking?

First of all, eye tracking is only supported on some variants of Somnium VR1 headsets (Visionary, Titan, Ultimate), otherwise you won't have access to eye tracking buttons or settings page.

If you have one of the supported headset variants, but eye tracking can't be activated or you see "Could not initialize camera stream." or "Unsupported frame format" errors, please follow these steps:

1) Check if the camera device is recognized

- Open Windows Device Manager and check that you see the "Somnium EyeCam" device in the Cameras category.
- If you don't see "Somnium EyeCam" device unplug the main headset USB from the PC, plug it in another USB3 port, and check Device Manager again.
- If you still don't see "Somnium EyeCam" device, then it could be a hardware issue with the camera device itself, so in that case please contact Somnium support directly.

2) Check if the camera data can be accessed

- Open a camera stream player (for example Camera software which is built into Windows).
- Select "Somnium EyeCam" as the stream source.
- You should get a full green image (due to custom image encoding).
- If you try aiming the headset displays toward a natural light source (window) you should also see some changes in the camera stream image (the image will still be green tinted).

3) Allow camera access in Windows privacy settings

- Open "Camera privacy settings" in Windows Settings.
- Make sure that "Allow access to camera on this device" is On.
- Make sure that "Allow apps to access your camera" is On.
- Make sure that "Allow desktop apps to access your camera" is On.

4) Send all Somnium logs to Somnium support (see "If I encounter any issues with the Somnium VR1 Tool, software, or headset itself, which files should I sent to Somnium support?")

- We are also currently preparing an additional debugging tool, which will be distributed to selected affected users to diagnose and resolve these issues.

- Which tracking system is supported by Somnium VR1, and do I need any additional accessories to be able to use the headset?

Somnium VR1 currently supports only lighthouse/SteamVR tracking system.

- The tracking sensors are embedded inside the Somnium VR1 headset, and you'll also need lighthouse/SteamVR base stations which act as a source of the tracking signals.
- For most standard setups we recommend using at least 2x lighthouse base stations v2.0. Some setups can work with only one base station or can work with the older v1.0 base stations, but the tracking quality will be limited.
- Any lighthouse compatible VR controller will work with Somnium VR1 headset. We recommend using the Valve Knuckles controllers, but also the Vive controllers or other third-party VR controllers are compatible. When using a new VR controller with Somnium VR1, first pair it through SteamVR - Devices - Pair Controller menu.

- Why are some buttons in Somnium VR1 Tool disabled?

Some buttons in Somnium VR1 Tool will be disabled / grayed out depending on the version of the headset you have and which application is currently running:

- If your Somnium VR1 headset doesn't have eye tracking support, the "Eye Tracking Calibration" and "Measure IPD" buttons will always be disabled, and you won't be able to access the Settings - Headset - Eyes page.
- If your Somnium VR1 headset doesn't have hand tracking support, you won't be able to access the Settings - Headset - Hands page.
- If your Somnium VR1 headset doesn't have mixed reality support, the "Mixed Reality - Start" button will always be disabled, and you won't be able to access the Settings - Mixed Reality page.

Also, most of the buttons in Connected Device section of the Somnium VR1 Tool will be disabled if no VR application is currently running

- If you don't see an application name in the bottom right corner of the Somnium VR1 Tool window it means that no app is currently running.
- If you've just started the end application (SteamVR, or any compatible OpenXR application), make sure to click back into the Somnium VR1 Tool window to refresh the button status.
- This includes "Eye Tracking Calibration", "Measure IPD", "Position Calibration", "VR View - Start", "Virtual Desktop - Start", and "Mixed Reality - Start" buttons.
- When no application is running you will also not see a current FPS value and pixels per eye resolution value.

- What is the optimal way to use Somnium VR1 Tool with DCS?

To fully utilize all Somnium VR1 functionality with DCS World, we recommend running it through our custom vrg/somnium OpenXR runtime.

Please follow these steps for basic setup:

- Turn off SteamVR (if SteamVR renders image into the headset DCS OpenXR will not be able to initialize properly).
- Alternatively, you can disable the SteamVR driver completely (either through Somnium VR1 Tool in Settings - Software - Drivers page, or through SteamVR in Startup/Shutdown - Manage Add-ons).
- Make sure that the Somnium OpenXR runtime is enabled in Somnium VR1 Tool (homepage or Settings - Software - Drivers page).

- Adjust the resolution and foveated resolution values in Somnium VR1 Tool (Settings - Headset - Rendering page).
- DCS will use quad views (foveated rendering) by default, so we recommend adjusting at least the main Resolution slider to lower values (for example 40-50%) and keeping the Foveated Rendering - Resolution slider around 100%
- Start DCS in VR mode.
- Reset your view position (UI Layer shortcuts - recenter VR Headset).

Additionally, if your Somnium headset has eye tracking support you can enable dynamic foveated rendering:

- Follow the same steps as mentioned above.
- Select if you want to use Simple or Advanced eye tracking calibration (Settings - Headset - Eyes page).
- Calibrate eye tracking (either with the button in Connected Device section or in Settings - Headset - Eyes page).
- Enable eye tracking (Settings - Headset - Eyes page).
- Alternatively, you can enable the AutoStart option (Settings - Headset - Eyes page), which will automatically start eye tracking after successful calibration.
- If the eye tracking is enabled, the foveated area in DCS will automatically move according to the eye tracking results.

- What is the optimal way to use Somnium VR1 Tool with MSFS?

To fully utilize all Somnium VR1 functionality with MSFS, we recommend running it through our custom vrg/somnium OpenXR runtime.

Please follow these steps for basic setup:

- Turn off SteamVR (if SteamVR renders image into the headset MSFS OpenXR will not be able to initialize properly).
- Alternatively, you can disable the SteamVR driver completely (either through Somnium VR Tool in Settings - Software - Drivers page, or through SteamVR in Startup/Shutdown - Manage Add-ons).
- Make sure that the Somnium OpenXR runtime is enabled in Somnium VR1 Tool (homepage or Settings - Software - Drivers page).
- Adjust the resolution value in Somnium VR1 Tool (Settings - Headset - Rendering page).
- MSFS doesn't support quad views (foveated rendering) so only the main resolution will apply (we recommend using 60%-100% depending on the performance requirements).
- Start MSFS.
- Enable the VR mode in MSFS (by default Ctrl+Tab).
- Reset view position (by default Space).

Additionally, if you want to enable spacewarp/reprojection functionality:

- Make sure that Spacewarp is enabled in Somnium VR1 Tool (Settings - Headset - Rendering page).
- Go to MSFS General options - Graphics - PC tab.
- Make sure that you have DirectX Version - DX11 selected (current implementation of spacewarp in vrg/somnium OpenXR runtime is not compatible with MSFS DX12).
- Go to MSFS General options - Graphics - VR tab.
- Switch "Reprojection mode" to "Depth.